

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG  
KHOA CÔNG NGHỆ THÔNG TIN**



**BÀI THI CUỐI KÌ  
XÂY DỰNG  
CÁC HỆ THỐNG NHÚNG**

**Hà Nội, 5/2021**

## **I. Đề 1**

### **A. Nêu các đặc tả của hệ thời gian thực:**

Để thỏa mãn tiêu chí thời gian thực, một hệ thống chỉ cần hoàn thành các công việc, các tác vụ trong một khoảng thời gian cho phép (**deadline**). Khoảng thời gian đó có thể là 1ns hay cũng có thể là hàng vài phút.

Một hệ thống thời gian thực có các đặc điểm tiêu biểu sau:

\* **Tính bị động:** Hệ thống phải phản ứng với các sự kiện xuất hiện vào các thời điểm thường không biết trước. Ví dụ, sự vượt ngưỡng của một giá trị đo, sự thay đổi trạng thái của một thiết bị quá trình phải dẫn đến các phản ứng trong bộ điều khiển.

\* **Tính nhanh nhạy:** Hệ thống phải xử lý thông tin một cách nhanh chóng để có thể đưa ra kết quả phản ứng một cách kịp thời. Tuy tính nhanh nhạy là một đặc điểm tiêu biểu, nhưng một hệ thống có tính năng thời gian thực không nhất thiết phải có đáp ứng thật nhanh mà quan trọng hơn là phải có phản ứng kịp thời đối với các yêu cầu, tác động bên ngoài.

\* **Tính đồng thời:** Hệ thống phải có khả năng phản ứng và xử lý đồng thời nhiều sự kiện diễn ra. Có thể, cùng một lúc một bộ điều khiển được yêu cầu thực hiện nhiều vòng điều chỉnh, giám sát ngưỡng giá trị nhiều đầu vào, cảnh giới trạng thái làm việc của một số động cơ.

\* **Tính tiên định:** Dự đoán trước được thời gian phản ứng tiêu biểu, thời gian phản ứng chậm nhất cũng như trình tự đưa ra các phản ứng. Nếu một bộ điều khiển phải xử lý đồng thời nhiều nhiệm vụ, ta phải tham gia quyết định được về trình tự thực hiện các công việc và đánh giá được thời gian xử lý mỗi công việc. Như vậy người sử dụng mới có cơ sở để đánh giá về khả năng đáp ứng tính thời gian thực của hệ thống.

### **B. Thế nào là định thời trong hệ thời gian thực?**

Bộ định thời là một loại đồng hồ chuyên dụng được sử dụng để đo khoảng thời gian. Định thời là khung thời gian thực ấn định cho một xử lý phải hoàn thành. Một bộ định thời đếm từ 0 trở lên để đo thời gian đã trôi qua thường được gọi là đồng hồ bấm giờ. Nó là một thiết bị đếm ngược từ một khoảng thời gian cụ thể và được sử dụng để tạo ra độ trễ thời gian, ví dụ: đồng hồ cát là một bộ đếm thời gian.

Bộ giám sát định thời (watchdog timer) là đồng hồ thời gian cứng (dùng các bộ đếm điện tử) với các ứng dụng sau đây:

+ Làm đồng hồ thời gian thực cho hệ thống.

- + Khởi động/khởi động lại một sự kiện sau một thời gian đặt trước.
- + Tạo khung cửa sổ thời gian cho một sự kiện.
- + Phân giải khoảng thời gian giữa hai sự kiện.

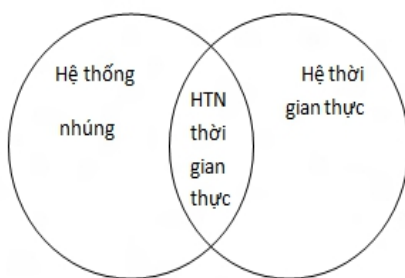
Bộ phận giám sát, hay đồng hồ thời gian mềm (lập các giá trị đếm cho một biến chương trình), các thao tác tăng/giảm giá trị đếm thực hiện bằng lệnh máy, do đó phụ thuộc vào CPU clock (mỗi loại CPU có clock khác nhau).

### **C. Thế nào là một hệ thống nhúng thời gian thực?**

Một hệ thống có khả năng thực hiện thời gian thực nghĩa là hệ thống đó phải thực hiện các chức năng của mình trong một khoảng thời gian xác định và nhỏ nhất có thể chấp nhận được. Khi đáp ứng được yêu cầu này, hệ thống đó có thể gọi là hệ thống thời gian thực

Các hệ thống này phải có khả năng đáp ứng các tín hiệu ngõ vào hoặc các sự kiện trong giới hạn một khoảng thời gian bắt buộc. Cho nên các hệ thống này không chỉ phải trả về một kết quả đúng mà còn phải nhanh nhất đáp ứng được yêu cầu về tốc độ của hệ thống. Trong các hệ thống thời gian thực, tốc độ cũng quan trọng không kém gì độ chính xác của nó.

Có 2 loại thời gian thực: thời gian thực cứng và thời gian thực mềm. Đối với hệ thống thời gian thực cứng, tất cả các chức năng của nó phải được thực thi chính xác trong một khoảng thời gian xác định, nếu không cả hệ thống sẽ bị lỗi nghiêm trọng. Ví dụ: hệ thống điều khiển không lưu. Đối với hệ thống thời gian thực mềm, các chức năng phải được thực hiện trong một khoảng thời gian xác định nhỏ nhất nhưng không bắt buộc.



### **D. Mô tả bằng hình vẽ và nguyên lý xử lý quản lý sự kiện thời gian thực**

Xử lý thời gian thực là hình thức xử lý thông tin trong một hệ thống để đảm bảo tính năng thời gian thực của nó. Như vậy, xử lý thời gian thực cũng có các đặc điểm tiêu biểu nêu trên như tính bị động, tính nhanh nhạy, tính đồng thời và tính tiên định. Để có thể phản ứng với nhiều sự kiện diễn ra cùng một lúc, một hệ thống xử lý thời gian thực sử dụng các quá trình tính toán đồng thời.

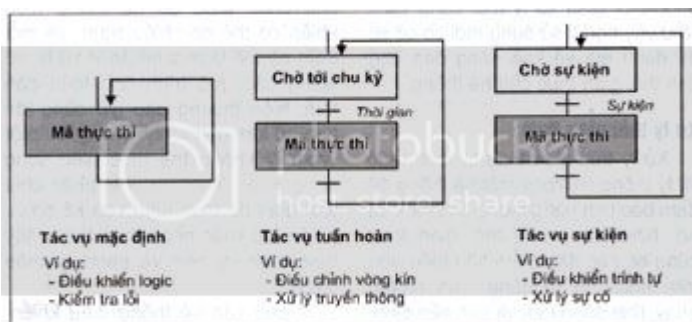
Quá trình tính toán là một tiến trình thực hiện một hoặc một phần chương trình tuần tự do hệ điều hành quản lý trên một máy tính, có thể tồn tại đồng thời với các quá trình khác kể cả trong thời gian thực hiện lệnh và thời gian xếp hàng chờ đợi thực hiện.

Các hình thức tổ chức các quá trình tính toán đồng thời:

- \* Xử lý cạnh tranh
- \* Xử lý song song
- \* Xử lý phân tán

Trong các hình thức trên đây thì hình thức xử lý cạnh tranh có vai trò chủ chốt. Mặc dù hệ thống điều khiển có thể có nhiều trạm, và mỗi trạm có thể là một hệ đa vi xử lý, số lượng các quá trình tính toán cần thực hiện thường bao giờ cũng lớn hơn số lượng vi xử lý. Trong khi một vi xử lý không thể thực hiện song song nhiều lệnh, nó phải phân chia thời gian để thực hiện xen kẽ nhiều nhiệm vụ khác nhau theo thứ tự tùy theo mức ưu tiên và phương pháp lập lịch.

Trong các hệ thống điều khiển, khái niệm tác vụ (task) cũng hay được sử dụng bên cạnh quá trình tính toán. Có thể nói, tác vụ là một nhiệm vụ xử lý thông tin trong hệ thống, có thể thực hiện theo cơ chế tuần hoàn (periodic task) hoặc theo sự kiện (event task). Các dạng tác vụ qui định trong chuẩn IEC 61131-3 (Programmable Controllers – Part3: Programming Languages) được minh họa trên hình 2. Ví dụ, một tác vụ thực hiện nhiệm vụ điều khiển cho một hoặc nhiều mạch vòng kín có chu kỳ trích mẫu giống nhau. Hoặc, một tác vụ có thể thực hiện nhiệm vụ điều khiển logic, điều khiển trình tự theo các sự kiện xảy ra. Tác vụ có thể thực hiện dưới dạng một quá trình tính toán duy nhất, hoặc một dãy các quá trình tính toán khác nhau.



## II. Đề 2:

**A. Đây là mô hình nguyên lý hoạt động của bộ định thời. Hãy giải thích cách hoạt động ?**

Bộ hẹn giờ cơ quan giám sát cơ bản có một giai đoạn hẹn giờ duy nhất, khi hết thời gian, thường sẽ đặt lại CPU

Watchdog về cơ bản là một bộ đếm, bắt đầu từ đếm số 0 và đạt đến một giá trị nhất định.

Watch timer có đầu ra nối vào chân RESET của CPU. Ban đầu bộ định thời được nạp một giá trị thời gian T, nếu sau T đơn vị thời gian này phần mềm vòng lặp không kết thúc, đầu ra của bộ định thời sẽ kích hoạt RESET, khởi động lại hệ thống. Nếu vòng lặp kết thúc với trạng thái bình thường, watchdog timer sẽ được nạp lại giá trị (khởi động lại watchdog) cho một chu kỳ mới.

Trong trường hợp phần mềm bị mắc kẹt vào vòng lặp vô tận, hệ thống sẽ không thể khởi động cơ quan giám sát do đó bộ đếm đạt đến giá trị nhất định và thiết lập lại hệ thống. Cơ quan giám sát ban đầu được tải với giá trị nhất định. Giá trị này được tính toán dựa trên thời gian chờ của cơ quan giám sát.

Dựa trên mỗi lần tăng bộ đếm nội bộ của cơ quan giám sát đánh dấu đồng hồ. Sau đó, có bộ so sánh so sánh giá trị đếm với giá trị đếm được tải (giá trị thời gian chờ) và nếu số lượng khớp với phần cứng cơ quan giám sát tạo và đặt lại tín hiệu.

### **B. Giả định viết một phần mềm vòng lặp xử lý với 3 tác vụ trong khung thời gian 40 $\mu$ s**

a. Mỗi tác vụ khác biệt được thực thi bởi một "main loop", nếu kết thúc hoàn hảo, đặt cờ trạng thái lên (Flagi set= TRUE).

Tất cả các main loop thực hiện tối đa trong 50 $\mu$ s. Sau vòng cuối cùng là đoạn mã kiểm tra. Định thời nạp lại giá trị và khởi động khi loop chạy lại

b. Nếu tất cả các Flags đều là True, khởi động chu kỳ watchdog mới(55 $\mu$ s), nếu Flags= FALSE, ghi nhận sự cố và đặt tất cả Flags=FALSE, watchdog không được khởi động lại(kick the dog) trong thời gian 55 $\mu$ s, đầu ra của bộ đếm watchdog sẽ kích hoạt RESET hệ thống.

c.

*Đoạn mã*

*If (all flag are OK)*

*{*

*Call(Reset Twd); //Hệ hoạt động bình thường,*

*//đặt mới giá trị 55 micro giây co watchdog*

*Jump mainloop; //Trở về chu kỳ mới;*

*}*

*else //nếu thực hiện đoạn code này có nghĩa*

*//counter sẽ vượt 55 giây ấn định xung đầu ra sẽ RESET CPU.*

*{*

*Log error; //Record failure*

```
Reset System; //WD reset CPU  
}
```

Dưới đây là giải thuật với định thời watchdog

**1. Twd>Ttask => đúng**

Ttask là thời gian thực hiện các tác vụ trong vòng lặp.

Twd là thời gian được nạp cho 1 chu kì, nếu vòng lặp kết thúc với thời gian lớn hơn

Twd thì đầu ra của watchdog timer sẽ kích hoạt RESET, khởi động lại hệ thống.

**2. Một hard deadline nên được đáp ứng.**

Nếu bất kì hard deadline nào bị bỏ lỡ thì hệ thống sẽ không đúng.

Yêu cầu một phương pháp để đảm bảo thời hạn được đáp ứng.

Hệ thống thời gian thực cứng là một hệ thống trong đó tất cả các tới hạn là khó.

Cách sử dụng:

Khi một máy tính là một phần của một hệ thống thời gian thực cứng, tất cả các phần mềm đang chạy trên nó có thể được điều chỉnh để đáp ứng thời hạn tất cả các hệ thống kiểm soát.

**3. Một soft deadline có thể được bỏ qua**

Hệ thống thời gian thực mềm là hệ thống thời gian thực trong đó một số thời hạn được coi nhẹ hơn.

ví dụ: thiết bị chuyển mạch điện thoại, ứng dụng đa phương tiện.

### III. Đề 3

Khi ghép với các thiết bị mà thiết bị đó cần thời gian để hoàn thành một xử lý trong quá trình ghép nối, CPU cần có ghi nhận trạng thái sẵn sàng của thiết bị để đánh giá và đưa ra quyết định tiếp theo. Quá trình kiểm tra trạng thái đôi khi phải thực hiện một vài lần trước khi quyết định tiếp tục hay từ bỏ trao đổi dữ liệu với thiết bị.

Để kiểm tra, nếu thiết bị trôi chảy thì sẽ thực hiện, nếu hỏng thiết bị không đợi sẵn sàng mãi được, chu trình phải kết thúc. Qui tắc do đó có điều kiện hay móc nối.

Cần ít nhất 2 cổng: 1 cổng để đọc trạng thái thiết bị, cổng kia để xác định dữ liệu vào hay ra.

Trạng thái thông thường ở mức độ đơn giản chỉ cần 1 bit để thực hiện. Có nhiều trường hợp trạng thái có thể là vài bit.

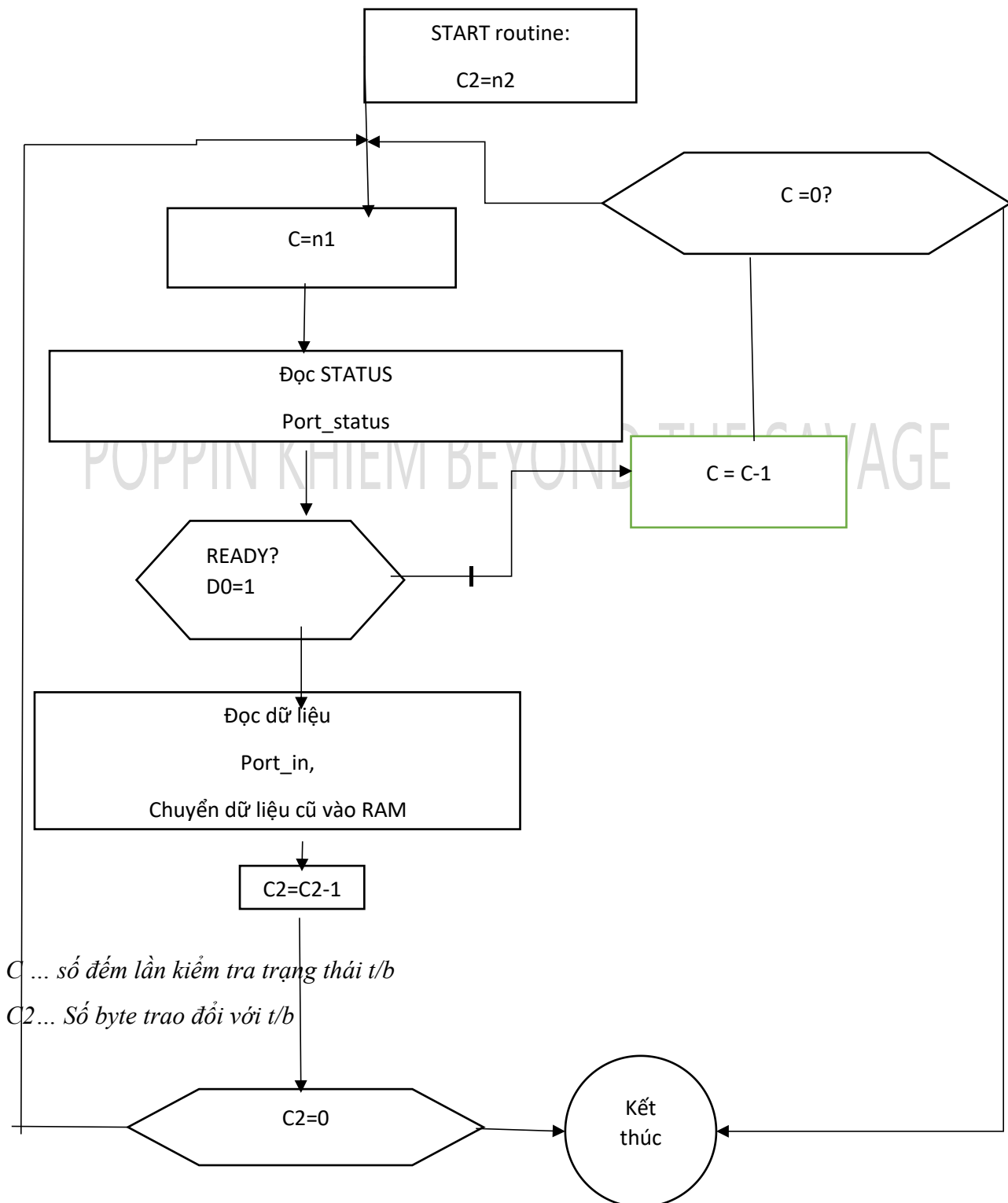
- Cần CS0/ để đọc trạng thái READY của thiết bị port\_status, giả định nối vào D0 của bit dữ liệu.

*Các bước thực hiện*

- CPU đưa địa chỉ port\_status ra BUS địa chỉ, vào giải mã, tạo CS0, đọc STATUS. Giá trị bit READY tại D0.

- Cần CS1/ để đọc dữ liệu qua port\_in hợp thành từ Flip/fop, cổng 3 state, nối vào bit D0 của BUS dữ liệu.
- CPU kiểm tra giá trị READY
- Nếu READY=0, quay lại đọc STATUS
- Nếu READY=1, CPU đọc dữ liệu vào ACC
- Thực hiện lệnh cất dữ liệu vào RAM

Lưu đồ điều khiển:



## IV. Đề 4

**Nêu kịch bản khởi động chạy phần mềm ở RAM sau khi mã đã copy từ ROM vào RAM:**

Ở kịch bản này, boot loader sẽ chuyển một chương trình nhỏ từ ROM vào RAM và kích hoạt nó chạy. Thường mã chương trình hệ thống trong ROM rất lớn mà được ghi kiểu nén để nạp vừa ROM, nên boot loader phải giải nén trước khi khởi động phần mã này và nó cần không gian nhớ ở RAM để thực hiện.

1) --- đến 6) giống như ở trên. Môi trường làm việc cho loader được khởi động ở RAM (3, 4, 5).

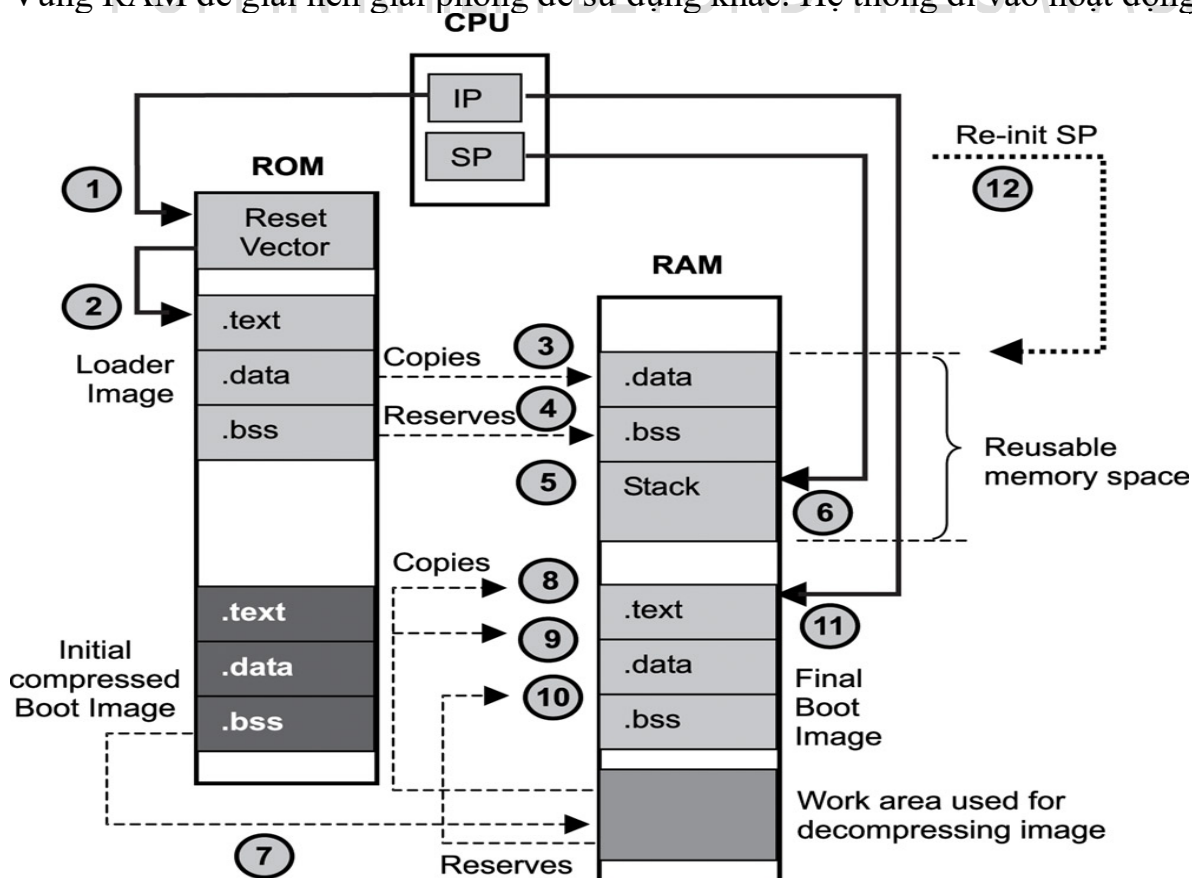
7) Loader copy phần mã nén của image vào RAM.

8) -> 10) Copy các phần mã đã giải nén vào các vùng làm việc tạm trong RAM (8, 9, 10). Hoàn tất giải nén image. Image trong RAM ở hình là đoạn Final Boot image.

10) ...

11) Loader chuyển điều khiển cho image bằng một lệnh JMP vào .text (nạp cho IP địa chỉ này trước khi JMP tới đó).

12) Vùng RAM mà loader chiếm khi được copy từ ROM là tái sử dụng, SP được tái khởi động để trở vào đó và được dùng như ngăn xếp cho một chương trình mới nào đó. Vùng RAM để giải nén giải phóng để sử dụng khác. Hệ thống đi vào hoạt động





*Youtube.com/Poppinkhiem*

**GOD WILLS IT  
DO NOT COPY WITHOUT  
PERMISSION**

POPPIN KHIEM BEYOND THE SAVAGE